

PATENT

Attorney Docket No: HP100111230-1
Express Mail Label No:EV 008783547 US

SYSTEMS AND METHODS FOR DETERMINING ACTIVITY FACTORS OF A
CIRCUIT DESIGN

RELATED APPLICATIONS

[0001] The present document contains material related to the material of copending, cofiled, U.S. patent applications Attorney Docket Number 100111221-1, entitled System And Method For Determining Wire Capacitance For A VLSI Circuit; Attorney Docket Number 100111227-1, entitled System And Method For Determining Applicable Configuration Information For Use In Analysis Of A Computer Aided Design; Attorney Docket Number 100111228-1, entitled Systems And Methods Utilizing Fast Analysis Information During Detailed Analysis Of A Circuit Design; Attorney Docket Number 100111232-1, entitled System And Method For Determining A Highest Level Signal Name In A Hierarchical VLSI Design; Attorney Docket Number 100111233-1, entitled System And Method For Determining Connectivity Of Nets In A Hierarchical Circuit Design; Attorney Docket Number 100111234-1, entitled System And Method Analyzing Design Elements In Computer Aided Design Tools; Attorney Docket Number 100111235-1, entitled System And Method For Determining Unmatched Design Elements In A Computer-Automated Design; Attorney Docket Number 100111236-1, entitled Computer Aided Design Systems And Methods With Reduced Memory Utilization; Attorney Docket Number 100111238-1, entitled System And Method For Iteratively Traversing A Hierarchical Circuit Design; Attorney Docket Number 100111257-1, entitled Systems And Methods For Establishing Data Model Consistency Of Computer Aided Design Tools; Attorney Docket Number 100111259-1, entitled Systems And Methods For Identifying Data Sources Associated With A Circuit Design; and Attorney Docket Number 100111260-1, entitled Systems And Methods For Performing Circuit Analysis On A Circuit Design, the disclosures of which are hereby incorporated herein by reference.

BACKGROUND

[0002] An electronic computer aided design (“E-CAD”) package uses an activity factor within a power analysis tool to determine switching power requirements associated with a signal of a Very Large Scale Integration (“VLSI”) circuit design. The activity factor defines the number of transitions the signal makes during one half of a clock cycle; the clock reference signal therefore has an activity factor of one. By determining activity factors for all signals in a VLSI circuit design, the power analysis tool may calculate switching power requirements for the entire VLSI circuit design. The switching power consumed by a particular signal is dependent on the frequency of transitions made by the signal and the capacitance of the signal net carrying the signal. The higher the signal frequency, the more often the net capacitance charges and discharges, increasing the switching power requirements.

[0003] The power analysis tool uses a vector logic simulator to simulate and determine the activity factor associated with each signal net of the VLSI circuit design. This simulation may take several hours or days since the VLSI circuit design typically has billions of engineering components. The delay associated with the simulation reduces productivity; continuous lost productivity due to lengthy engineering development slows technology advancement and can result in significant costs, as well as lost business.

[0004] To reduce the time period associated with determining switching power requirements of the VLSI circuit design, an average activity factor is often estimated and applied to all signal nets of the VLSI circuit design, thereby removing the need to separately calculate activity factors for each signal net. This estimate reduces the number of calculations required in determining switching power requirements of the VLSI circuit design, but it also reduces the accuracy of the results.

SUMMARY

[0005] In one embodiment, a method determines activity factors of a circuit design. An activity factor is assigned to one or more node types. One or more signal nets from a netlist of the circuit design are read. The signal nets are processed to associate one of the node types with each of the signal nets. An activity factor is determined for each of the signal nets based upon the node type.

[0006] In another embodiment, a system determines activity factors of a circuit design. A circuit recognition tool is responsive to control by an E-CAD tool to determine node types of one or more signal nets of the circuit design. Memory stores activity factors associated with the node types, such that the activity factor is determined by node type. An analysis tool is operable to access the memory to determine an activity factor for each of the signal nets based upon node type.

[0007] In another embodiment, a system determines activity factors of a circuit design, including: means for reading one or more signal nets from a netlist of the circuit design; means for processing the signal nets to determine a node type for each of the signal nets; and means for determining an activity factor for each of the signal nets based upon node type.

[0008] In another embodiment, a software product has instructions, stored on computer-readable media, wherein the instructions, when executed by a computer, perform steps for determining activity factors of a circuit design, including: reading one or more signal nets from a netlist of the circuit design; processing the signal nets to associate a node type with each of the signal nets; and determining an activity factor for each of the signal nets based upon the node type.

BRIEF DESCRIPTION OF THE FIGURES

[0009] FIG. 1 is a block diagram illustrating one system for determining activity factors of a circuit design.

[0010] FIG. 2 is a block diagram illustrating exemplary analysis of a netlist to determine (a) node types for signal nets within the netlist and (b) activity factors associated with the signal nets.

[0011] FIG. 3 is an exemplary schematic diagram of one block of a circuit design illustrating signal nets with four different node types.

[0012] FIG. 4 is a flowchart illustrating one process for determining activity factors of a circuit design.

[0013] FIG. 5 is a flowchart illustrating one process for using activity factors to determine switching power requirements.

DETAILED DESCRIPTION OF THE FIGURES

[0014] FIG. 1 is a block diagram illustrating one system 100 for determining activity factors of a circuit design (e.g., circuit design 116). As described below, such activity factors are for example useful in determining switching power requirements associated with part or all of the circuit design. System 100 has a computer 102 with a computer memory 104, a processor 106, a storage unit 108 and a user interface 110. Storage unit 108 may for example be a disk drive that stores programs and data of computer 102. Storage unit 108 is illustratively shown storing an E-CAD tool 114, circuit design 116, and an activity factor lookup table 124. Circuit design 116 is, for example, a very-large-scale integrated (“VLSI”) circuit design created by E-CAD tool 114. E-CAD tool 114 additionally has an analysis tool 120 and a circuit recognition tool 122.

[0015] Circuit design 116 includes a netlist 118, which defines the signal nets that interconnect various design elements of circuit design 116. A “design element” is for example a capacitor, a logic gate, a resistor, a terminal, a transistor, etc. A “signal net” is a single electrical path in a circuit that has the same electrical characteristics at all of its points. Any collection of wires that carries the same signal between design elements is a signal net. If the design elements allow the signal to pass through unaltered (as in the case of a terminal), then the signal net continues on subsequently connected wires. If, however, the design elements modify the signal (as in the case of a transistor or logic gate), then the signal net terminates at that design element and a new signal net begins on the other side.

[0016] With further regard to FIG. 1, processor 106 loads E-CAD tool 114, circuit design 116 and activity factor lookup table 124 from storage unit 108 into computer memory 104, such that E-CAD tool 114 is executable by processor 106 and has access to circuit design 116 and activity factor lookup table 124, as shown. Analysis tool 120 and circuit recognition tool 122 are similarly loaded into computer memory 104. User interface 110 connects to a terminal 112 (e.g., a keyboard) external to computer 102. Through terminal 112 and user interface 110, a design engineer interacts with and controls E-CAD tool 114 (and, likewise, analysis tool 120 and circuit recognition tool 122) to perform various functions. The design engineer may

for example instruct E-CAD tool 114 to use analysis tool 120 to determine switching power requirements for circuit design 116, as discussed below.

[0017] By way of illustrative operation, a design engineer instructs E-CAD tool 114, and hence analysis tool 120, to determine switching power requirements for circuit design 116. Once initiated, analysis tool 120 performs calculations, for example employing Equation 1 below, to determine the switching power for one or more signal nets of netlist 118. Equation 1 illustrates one calculation utilizing an activity factor to determine switching power required by one signal net of circuit design 116.

[0018] EQUATION 1

*Switching Power = Activity Factor * Net Capacitance * Volts² * Clock Frequency,*
where *Switching Power* is the switching power required by the signal net, *Net Capacitance* is the capacitance of the signal net for which the power is being calculated, *Volts* is the voltage difference through which the signal net switches, and *Clock Frequency* is the frequency of the clock on which the activity factor is based.

[0019] Analysis tool 120 utilizes circuit recognition tool 122 to determine node type for one or more signal nets of netlist 118. Circuit recognition tool 122 processes the signal nets to determine a node type that most closely resembles the signal net. For example, a node type may be static, dynamic, clock or non-toggling. Each node type has an associated activity factor; analysis tool 120 utilizes the node type to determine the activity factor for each signal net without separately simulating the signal net. Analysis tool 120 then utilizes the activity factors to characterize circuit design 116, for example allowing the design engineer to estimate switching power requirements of circuit design 116 without undo delay.

[0020] Table 1 is one example of activity factors assigned to the following node types: static, dynamic, clock and non-toggling. The activity factor for each node type may be determined from the results of a simulation performed, for example, on a vector based switch level simulator. Other node types (“other”, as shown) may be defined to further increase accuracy in determining switching power requirements, as a matter of design choice.

<u>Node type</u>	<u>Factor</u>
Static	0.1
Dynamic	0.2
Clock	1.0
Non-Toggling	0.0
Other	TBD

TABLE 1

[0021] FIG. 2 is a block diagram illustrating an exemplary analysis of a netlist 118' to determine (a) node types for signal nets defined by netlist 118' and (b) activity factors associated with the signal nets. In FIG. 2, netlist 118' is illustratively shown with six nets: net A, net B, net C, net D, net E and net F. Analysis tool 120 instructs circuit recognition tool 122 to process one or more signal nets of netlist 118'. Circuit recognition tool 122 determines a node type for each signal net. Analysis tool 120 uses this determined node type to retrieve an activity factor from activity factor lookup table 124. In one illustrative example, analysis tool 120 uses this activity factor to determine switching power requirements for selected signal nets A-F of netlist 118'; these requirements may be summed to estimate switching power requirements for some or all of circuit design 116.

[0022] In one illustrative example, analysis tool 120 reads signal net A from netlist 118', via data path 134, and stores information of signal net A in table 130, via data path 140, as shown. Analysis tool 120 instructs circuit recognition tool 122, via control path 136, to process signal net A. Circuit recognition tool 122 reads signal net A from netlist 118', via data path 138. Circuit recognition tool 122 processes signal net A to determine that signal net A is a static node type and informs analysis tool 120 of the static node type via data path 141. Analysis tool 120 then updates table 130, via data path 142, with the node type. Analysis tool 120 uses the static node type to retrieve an activity factor of 0.1 from activity factor lookup table 124, via data path 144, and updates table 130 via data path 146, as shown. Analysis tool 120 reads the activity factor from table 130 via data path 148 to determine and output (as indicated by arrow 149) a switching power requirement for signal net A. Analysis tool 120 and circuit recognition tool 122 process each selected signal net of

netlist 118', as directed by the design engineer, to complete table 130. In this example, circuit recognition tool 122 determines that signal net B is a dynamic node type, signal net C is a clock node type, signal net D is a non-toggling node type, signal net E is a static node type and signal net F is a clock node type. Continuing with the example of Table 1, signal net B has an activity factor of 0.2, signal net C has an activity factor of 1.0, signal net D has an activity factor of 0.0, signal net E has an activity factor of 0.1 and signal net F has an activity factor of 1.0. Analysis tool 120 may then output results 149 with switching power requirements for all signal nets A-E.

[0023] Upon reading and fully appreciating this disclosure, those of ordinary skill in the art appreciate that Table 1 and FIG. 2 are shown illustratively, and should not be limiting. For example, in one embodiment, analysis tool 120 acquires the node type, for each net, directly from circuit recognition tool 122 and then associates activity factors with the net to determine switching power requirements. In another example, activity factor lookup table 124 may exist in a different form, with like function, without departing from the scope hereof; for example table 124 may exist as data structures within system 100, FIG. 1. Similarly, in another example, those skilled in the art appreciate that table 130 may exist in different form, with like function, without departing from the scope hereof. By way of example, in one embodiment table 130 is functionally replaced by data structures within analysis tool 120 or system 100, FIG. 1. Accordingly, the rows of table 130 may illustratively define the data structures to associate each net (e.g., net A) with its node type (e.g., "static") and its activity factor (e.g., 0.1).

[0024] A significant characteristic of VLSI and other types of circuit design is a reliance on hierarchical description. A primary reason for using hierarchical description is to hide the vast amount of detail in circuit design 116. By reducing the distracting detail to a single object that is lower in the hierarchy, one can greatly simplify many E-CAD operations. For example, simulation, verification, design-rule checking, and layout constraints can all benefit from hierarchical representation, which makes them more computationally tractable. Since many circuits are too complicated to be easily considered in their totality, a complete design is often viewed as a collection of component aggregates that are further divided into

sub-aggregates in a recursive and hierarchical manner. In VLSI circuit design, these aggregates are commonly referred to as blocks (or cells). The use of a block at a given level of hierarchy is called an ‘instance’.

[0025] FIG. 3 is an exemplary schematic diagram of one part 150 of circuit design 116, illustrating signal nets with four different node types. Part 150 has three blocks 154, 156 and 158, and five signal nets 160, 162, 164, 166 and 168. Block 154 is a clock generator that outputs a clock signal over signal net 162. Block 156 is a counter that inputs the clock signal from signal net 162 and outputs a count signal on signal net 164. Block 158 is a decoder that receives and decodes the counter signal via signal net 164 and outputs a decoded signal onto net 166. Signal net 160 and signal net 168 represent power supply connections (VDD and GND, respectively) for blocks 154, 156 and 158, and exemplify non-toggling node types. Signal net 162 is an example of a clock node type; signal net 164 is an example of a dynamic node type; signal net 166 is an example of a static node type. Circuit recognition tool 122 may utilize characteristics (e.g., signal name) of signal nets 160, 162, 164, 166 and 168 and connected design elements to determine and associate a node type with each net. In one example, circuit recognition tool 122 determines that signal net 162 is named as “clock signal” in the netlist and accordingly associates a clock type node to signal net 162.

[0026] FIG. 4 is a flowchart illustrating one process 200 for determining activity factors of circuit design 116. Process 200 is, for example, implemented by circuit recognition tool 122 when an E-CAD tool determines activity factors for one or more signal nets of a netlist (e.g., netlist 118). In step 202, circuit recognition tool 122 reads a signal net (e.g., signal net A) from the netlist. In step 204, circuit recognition tool 122 processes the signal net read in step 202 to determine a node type. The node type is stored or made available as output, indicated by arrow 206, for use by analysis tool 120. Using the example of FIG. 2 and Table 1, in step 204, circuit recognition tool 122 determines that signal net A is a static node type.

[0027] Steps 202 and 204 repeat as necessary or desired to process other signal nets within the netlist. By way of example, all signal nets of the netlist are processed to determine node types for all of design 116.

[0028] FIG. 5 is a flowchart illustrating one process 300 that utilizes the node types determined from process 200, for example to determine switching power requirements of circuit design 116. Process 300, for example, is implemented by analysis tool 120. In step 302, analysis tool 120 reads information of a signal net (e.g., signal net A) from the netlist. In step 304, analysis tool 120 reads the node type (arrow 306) determined in step 204 of process 200. Process 300 may receive the activity factor directly from circuit recognition tool 122 (arrow 206), or, for example, from one or more data structures such as set forth in table 130, FIG. 2.

[0029] In step 308, analysis tool 120 determines an activity factor based upon the node type read in step 304. In the example of FIG. 2, analysis tool 120 utilizes Table 1 to determine that signal net A has an activity factor of 0.1, based upon the ‘static’ node type. In step 310, analysis tool 120 uses the activity factor, determined in step 308, to perform analysis calculations (e.g., switching power requirements) for the signal net read in step 302. Equation 1 may be used in step 310. The calculation results are output, as indicated by arrow 312, and may be stored or printed for later use. Steps 302, 304, 308 and 310 repeat as necessary or desired to process other signal nets within the netlist. By way of example, all signal nets of netlist 118 are processed to determine switching power requirements for all of design 116, as output 312.

[0030] Changes may be made in the above methods and systems without departing from the scope hereof. It should thus be noted that the matter contained in the above description or shown in the accompanying drawings should be interpreted as illustrative and not in a limiting sense. The use of the activity factors by the power analysis tool is given as an example. Activity factors may also be used in other processes while remaining within the scope of the present method and system. The following claims are intended to cover all generic and specific features described herein, as well as all statements of the scope of the present method and system, which, as a matter of language, might be said to fall there between.